

Channel Decoding with a Bayesian Equalizer

Luis Salamanca, Juan José Murillo-Fuentes
DTSC University of Seville,
Camino de los Descubrimientos s/n,
41092 Seville, Spain.
{salamanca,murillo}@us.es

Fernando Pérez-Cruz
DTSC University Carlos III in Madrid,
Avda. de la Universidad 30, 28911,
Leganés (Madrid), Spain.
fernando@tsc.uc3m.es

Abstract—Low-density parity-check (LDPC) decoders assume the channel estate information (CSI) is known and they have the true a posteriori probability (APP) for each transmitted bit. But in most cases of interest, the CSI needs to be estimated with the help of a short training sequence and the LDPC decoder has to decode the received word using faulty APP estimates. In this paper, we study the uncertainty in the CSI estimate and how it affects the bit error rate (BER) output by the LDPC decoder. To improve these APP estimates, we propose a Bayesian equalizer that takes into consideration not only the uncertainty due to the noise in the channel, but also the uncertainty in the CSI estimate, reducing the BER after the LDPC decoder.

I. INTRODUCTION

Single input single output (SISO) communication channels can be characterized by a linear finite impulsive response that either represents the dispersive nature of a physical media or the multiple paths of wireless communications [1]. This representation causes inter-symbol interference (ISI) at the receiver end that can impair the digital communication. Given this channel estate information (CSI), the maximum likelihood sequence detector (MLSD) [2] -Viterbi Algorithm- provides the optimal transmitted sequence at the receiver end. And, if we are interested in a posteriori probabilities (APP) for the transmitted symbols, we can use the BCJR algorithm [3], that provides bitwise optimal decisions.

Channel encoders introduce controlled redundancy in the transmitted sequence to correct errors caused by the channel. Modern channel decoders, such as turbo or low-density parity-check (LDPC) codes [4], need accurate APP estimates to be able to achieve channel capacity [5].

In a previous work, we have shown that accurate APP estimates increase the performance of LDPC decoders [6], although that work focuses on nonlinear channel estimation.

In this paper, we study how the uncertainty in the estimation of the CSI affects the optimal performance of modern channel decoders. The CSI is acquired using a training sequence [1] and it is typically estimated by maximum likelihood (ML). These training sequences are necessary short to reduce the transmission of non-informative symbols, yielding inaccurate CSI estimates. Thus, the BCJR assuming an ML estimation (hereafter refers as ML-BCJR) only delivers an approximation to the APP for each symbol because it does not include the uncertainty in the estimate. Inaccuracies in the APP estimates degrade the performance of channel decoders for turbo or LDPC codes [7], [8]. We consecutively propose and analyze a

Bayesian equalizer, which takes into account the uncertainty in the CSI estimate to produce more accurate APP estimates.

The difference between the bit error rate (BER) of the ML-BCJR and the proposed Bayesian equalizer is not significant, although it slightly favors the Bayesian equalizer. However, this is not an accurate measure of the quality of the APP estimates for each equalizer since it only considers hard decisions, in contrast to the soft inputs needed by modern channel decoders. Thus, assuming LDPC coding in our communication system, we can compare the quality of the APP estimates for each equalizer. We experimentally show that at the output of the LDPC decoder the Bayesian equalizer considerably improves the performance of the ML-BCJR equalizer, when we measure the probability of error. These gains are more significant for high signal to noise ratios, channels with long impulsive responses and/or short training sequences.

There are some related works where uncertainties are exploited on the estimation of the transmitted symbols. In the framework of turbo-receivers [9], some approaches can be found in the literature that incorporate these uncertainties in the iterative process of equalization and decoding. It has been exploited in [10], where the authors use an MMSE to estimate the channel, and in [11], [12], where they do not focus on the optimal estimation of the APP. In [13] we find a proposal to estimate some parameters in a OFDM system to later include them in the decoding. In [14], the authors consider the channel estimation inaccuracies during the decoding process by means of a practical decoding metric.

The paper is organized as follows. In Section II we describe the structure of a SISO communication system and the ML-BCJR solution. The proposed Bayesian equalization technique is presented in Section III. Experimental results in Section IV help to illustrate the performance of our method. Finally, Section V ends with conclusions and some proposals for future work.

II. ML-BCJR EQUALIZATION

We consider the discrete-time dispersive communication system depicted in Fig. 1. The channel $H(z)$ is completely specified by the CSI, i.e., $\mathbf{h} = [h_1, h_2, \dots, h_L]^T$, where L is the length of the channel. We model the values of the channel \mathbf{h} as independent Gaussians with zero-mean and variance equal to $1/L$ (Rayleigh fading).

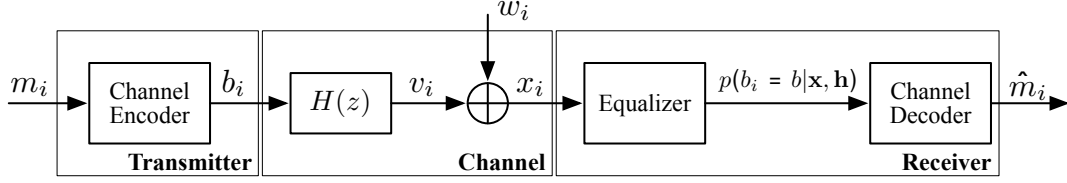


Fig. 1. System model.

A block of K message bits, $\mathbf{m} = [m_1, m_2, \dots, m_K]^\top$, is encoded with a rate $R = K/N$ to obtain the codeword $\mathbf{b} = [b_1, b_2, \dots, b_N]^\top$ that is transmitted over the channel using a BPSK modulation:

$$x_i = \mathbf{b}_i^\top \mathbf{h} + w_i, \quad (1)$$

where $\mathbf{b}_i = [b_i, b_{i-1}, \dots, b_{i-L+1}]^\top$ and w_i is additive white Gaussian noise (AWGN) with variance σ_w^2 . Thus, the received sequence is $\mathbf{x} = [x_1, x_2, \dots, x_N]^\top$.

At the beginning of every block we transmit a preamble with n known bits ($b_1^\circ, \dots, b_n^\circ$) and the receiver uses $\mathcal{D} = \{x_i^\circ, b_i^\circ\}_{i=1}^n$, the training sequence, to estimate the channel. The ML criterion is widely considered for the task of estimation:

$$\hat{\mathbf{h}}_{ML} = \arg \max_{\mathbf{h}} p(\mathbf{x}^\circ | \mathbf{b}^\circ, \mathbf{h}). \quad (2)$$

Once we have estimated the channel coefficients with the preamble, we apply the BCJR algorithm to obtain the posterior probability estimates for each transmitted bit:

$$p(b_i = b | \mathbf{x}, \hat{\mathbf{h}}_{ML}) \quad i = 1, 2 \dots N. \quad (3)$$

Finally we decode the received word using the LDPC decoder to obtain a maximum a posteriori estimate for m_i .

The LDPC decoder relies on the estimates in (3) being accurate and, if they are not, the decoding might not finish or might return an incorrect codeword. In Fig. 2 we show $p(b_i = 1 | \mathbf{x}, \hat{\mathbf{h}}_{ML})$ versus the probability $p(b_i = 1 | \mathbf{x}, \mathbf{h})$ for one estimation of the fading channel $\mathbf{h} = [1, 0.1]$ obtained assuming $n = 6$, and a signal to noise ratio (SNR) equal to 0 dB. We can see in Fig. 2 that the predictions for each bit are quite accurate. But these posterior probability estimates are overconfident roughly half of the time, because ML is an unbiased estimator of the CSI, which could derail the LDPC decoder, because a bit with high confidence for a zero or a one is hard to overrule if it is incorrect.

III. BAYESIAN EQUALIZATION

If we provide the BCJR with the true CSI we have the exact APP [3]. However, in practice we usually use the ML criterion in (2) to estimate the CSI from a training sequence. This estimated CSI, provided as ground truth to the BCJR algorithm, assigns inaccurate estimates of the APP and might mislead the channel decoder to deliver the incorrect codeword (or not to converge at all). We propose a Bayesian equalizer that takes into account both the uncertainty about the noise

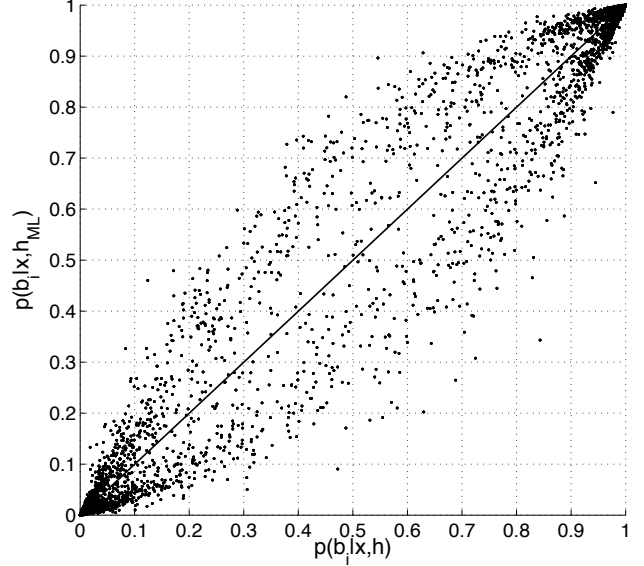


Fig. 2. Calibration curve for the ML-BCJR assuming SNR = 0 dB and $\mathbf{h} = [1, 0.1]$.

and the uncertainty about the CSI estimate. We compute the posterior probability as:

$$p(b_i = b | \mathbf{x}, \mathcal{D}) = \int p(b_i = b | \mathbf{x}, \mathbf{h}) p(\mathbf{h} | \mathcal{D}) d\mathbf{h}, \quad (4)$$

where $b = \pm 1$ and

$$\begin{aligned} p(\mathbf{h} | \mathcal{D}) &= \frac{p(\mathbf{h}) p(\mathbf{x}^\circ | \mathbf{b}^\circ, \mathbf{h})}{p(\mathbf{x}^\circ | \mathbf{b}^\circ)} \\ &= \frac{p(\mathbf{h}) \prod_{i=1}^n p(x_i^\circ | b_i^\circ, \mathbf{h})}{p(x_1^\circ, \dots, x_n^\circ | b_1^\circ, \dots, b_n^\circ)}, \end{aligned} \quad (5)$$

is the posterior probability for the CSI, given the likelihood (Gaussian noise) and the prior (e.g. Rayleigh fading).

The marginalization of \mathbf{h} in (4) provides equal or better APP estimates than the ML-BCJR, because it includes all the information of $p(\mathbf{h} | \mathcal{D})$. In fact, if the training sequence is large enough, this Gaussian posterior tends to a multidimensional delta centered at its mean, whose value tends to the real value of \mathbf{h} , as $\hat{\mathbf{h}}_{ML}$ does. However, the performance of the ML-BCJR is misled in case of uncertainty in the CSI. Conversely, the Bayesian equalizer in (4) considers the uncertainty in the estimation, providing more accurate APP estimates.

This computation of the APP does not yield a significant improvement in the BER after the equalizer, because for

detection we only consider if the probability is lower or higher than 0.5. This is illustrated in Section IV.

However, the main advantage of the Bayesian equalizer is that the performance of the soft-decoder improves with this new estimation of the probabilities, that translates into better results in terms of BER at the output of the decoder. In Fig. 3 we show $p(b_i = 1|\mathbf{x}, \mathcal{D})$ versus $p(b_i = 1|\mathbf{x}, \mathbf{h})$ for a fading channel $\mathbf{h} = [1, 0.1]$ estimated with the same training sequences assumed in Fig 2. Compared to the ML-BCJR in Fig. 2, in the Bayesian equalizer the zero-mean Gaussian prior tends to underestimate the CSI. Then, the averaging over all possible values of $p(\mathbf{h}|\mathcal{D})$ gives mostly underconfident posterior probability estimates, as we can observe in Fig. 3. Therefore, the LDPC decoder does not have such a strong preference for the received bits and they can be easily flipped, if necessary.

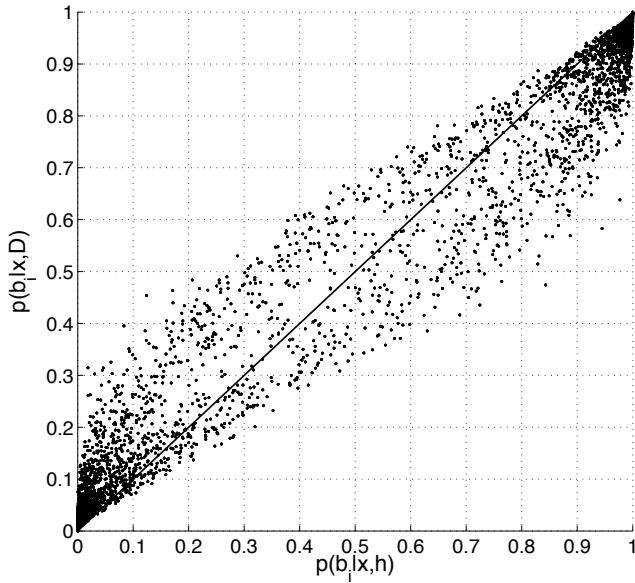


Fig. 3. Calibration curve for the Bayesian equalizer assuming SNR = 0 dB and $\mathbf{h} = [1, 0.1]$.

A. Computation of the solution

We propose to use Monte Carlo to obtain an approximate result to (4) considering the following steps:

- 1) Calculate the posterior of the channel: in (5) the numerator is the product of the likelihood $p(\mathbf{x}^\circ|\mathbf{b}^\circ, \mathbf{h})$ and the prior of \mathbf{h} . Considering a system with BPSK modulation in a Rayleigh channel, both terms are real valued Gaussians distributed as:

$$p(\mathbf{x}^\circ|\mathbf{b}^\circ, \mathbf{h}) \sim \mathcal{N}((\mathbf{B}^\circ)^\top \mathbf{h}, \sigma_w^2 \mathbf{I}), \quad (6)$$

$$p(\mathbf{h}) \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad (7)$$

where $\mathbf{B}^\circ = [\mathbf{b}_1^\circ, \mathbf{b}_2^\circ, \dots, \mathbf{b}_n^\circ]$ is an $L \times n$ matrix, and without loss of generality we assume that the variance of the CSI is equal to one. The expressions for the mean and the covariance matrix of the posterior when

both terms are Gaussians are straightforward [15]. These expressions can be particularized for our system as:

$$\bar{\mathbf{h}}_{\mathbf{h}|\mathcal{D}} = (\mathbf{I} + \mathbf{B}^\circ (\mathbf{B}^\circ)^\top \sigma_w^{-2})^{-1} \mathbf{B}^\circ \sigma_w^{-2} \mathbf{x}^\circ, \quad (8)$$

$$\mathbf{C}_{\mathbf{h}|\mathcal{D}} = (\mathbf{I} + \mathbf{B}^\circ (\mathbf{B}^\circ)^\top \sigma_w^{-2})^{-1}, \quad (9)$$

where $\mathbf{x}^\circ = [x_1^\circ, x_2^\circ, \dots, x_n^\circ]^\top$.

- 2) Produce random samples from the posterior: with the vector of means and the covariance matrix, we can sample to obtain M random samples.
- 3) Solve the BCJR algorithm: the APP estimates of each transmitted bit is computed for the M different samples of $p(\mathbf{h}|\mathcal{D})$.
- 4) Computation of (4): the M different values of $p(b_i = b|x_1, \dots, x_N, \mathbf{h})$ average the APP of each transmitted bit over all possible cases of \mathbf{h} :

$$\begin{aligned} p(b_i = b|x_1, \dots, x_N, \mathcal{D}) &\approx \\ &\approx \frac{1}{M} \sum_{j=1}^M p(b_i = b|x_1, \dots, x_N, \mathbf{h}_j), \end{aligned} \quad (10)$$

which yields an approximation of (4).

As already discussed, if we use these optimal APP estimates as inputs to a soft-decoder, the system achieves better performance, especially in case of inaccurate estimations of the CSI, i.e., the length of the training sequence is short compared to the number of channel taps. However, this solution is computationally demanding, because we have to calculate M times the BCJR algorithm, whose complexity increases exponentially with L .

IV. SIMULATION RESULTS

To illustrate the performance of the proposed receiver, we compare its bit error rate curves to the ones of the ML-BCJR, before and after the decoder. In all experiments we consider the following scenario:

- Block frames of 500 random bits encoded with a regular LDPC code (3,6) of rate 1/2.
- Up to 10000 frames of 1000 bits are transmitted over the channel.
- Between frames, a training sequence of n random uncoded bits is transmitted to estimate the channel.
- Every frame, and its associated training sequence, is sent over the same Rayleigh fading channel. We consider that the channel coherence time is greater than the duration of the frame, i.e., the channel does not change during this time. Furthermore, in our experiments we take the same value for the taps of the channel during all transmitted frames. For the channel with three taps $-L = 3$ - these values are:

$$H(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2},$$

and for $L = 6$:

$$\begin{aligned} H(z) &= 0.1600 + 0.5450z^{-1} - 0.6720z^{-2} + \\ &+ 0.2560z^{-3} + 0.0950z^{-4} - 0.3890z^{-5}. \end{aligned}$$

- We consider for the Bayesian estimation a prior with zero mean and variance equal to 1 for all the taps.

We depict in Fig. 4 the BER measured after the ML-BCJR (dashed line) and Bayesian (solid line) equalizers for the channel with $L = 3$ and different lengths of the training sequence. The differences between both equalizers are negligible, although the BER of the Bayesian equalizer is always lower.

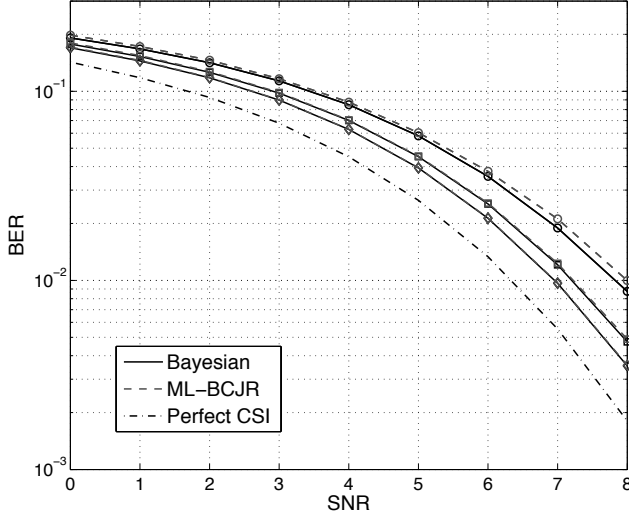


Fig. 4. BER performance for Bayesian equalizer (solid lines) and ML-BCJR (dashed lines) before the decoder, for a channel with 3 taps and different lengths of the training sequence, $n = 10$ (o), $n = 15$ (□) and $n = 20$ (◇). Dash-dotted line illustrates performance assuming perfect CSI.

In Fig. 5 and 6 we show the BER after the LDPC decoder has corrected the errors introduced by the channel. In these plots, we can observe that for short training sequences the gains of using the Bayesian equalizer over the ML-BCJR are significant (over 1 dB). The difference between the results in Fig. 4, and Fig. 5 and 6 can be explained by the APP estimates given by each procedure. When we measure the BER after the equalizer (Fig. 4), we only care if each bit has been correctly decoded and, consequently, we are only measuring how good the APP estimate of the 50% percentile is. When we measure the BER after the LDPC decoder (Fig. 5 and 6), we need that the APP estimates are accurate everywhere, because the LDPC decoder needs these estimates to decode correctly the transmitted codeword, i.e., the Belief Propagation algorithm uses the APP for each individual bit. These results sustain our claim that the Bayesian equalizer provides more accurate predictions of the APP than the ML-BCJR equalizer, as the LDPC decoder is able to better decode with them.

Particularly, in Fig. 5 we can observe that the Bayesian equalizer improves the ML-BCJR equalizer as we increase the SNR for a fixed length of the training sequence and both equalizers tend to coincide as we increase the length of the training sequence. This last result is expected, because as we increase the length of the training sequence the posterior for the CSI tends to a delta function and the ML-BCJR and Bayesian equalizers coincides. But for these equalizers to

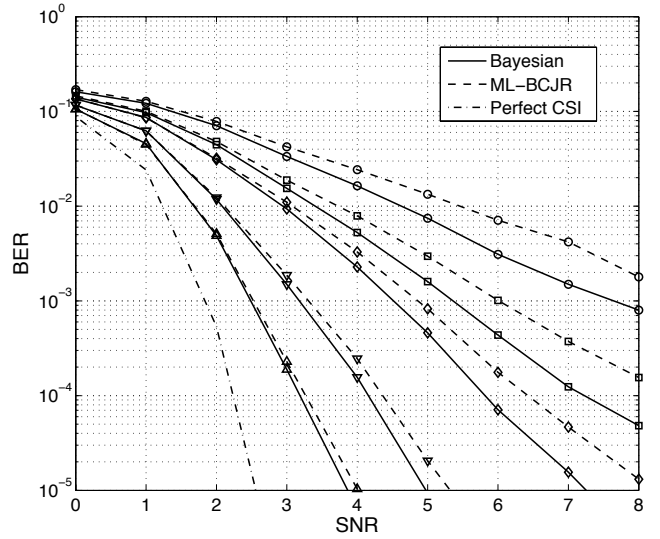


Fig. 5. BER performance for Bayesian equalizer (solid lines) and ML-BCJR (dashed lines) after the LDPC decoder, for a channel with 3 taps and different lengths of the training sequence, $n = 10$ (o), $n = 15$ (□), $n = 20$ (◇), $n = 35$ (▽) and $n = 60$ (△). Dash-dotted line illustrates performance assuming perfect CSI.

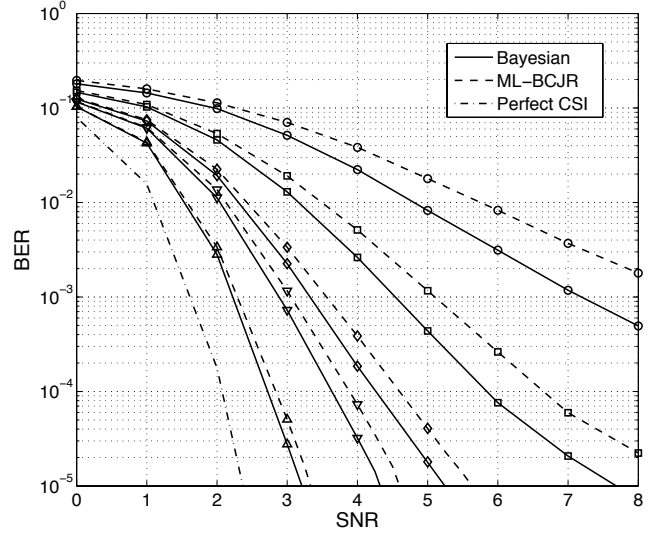


Fig. 6. BER performance for Bayesian equalizer (solid lines) and ML-BCJR (dashed lines) after the decoder, for a channel with 6 taps and different lengths of the training sequence, $n = 15$ (o), $n = 25$ (□), $n = 40$ (◇), $n = 50$ (▽) and $n = 90$ (△). Dash-dotted line illustrates performance assuming perfect CSI.

coincide, we need over 60 training samples (more than 20 per coefficient) and for shorter training sequences the Bayesian equalizer is vastly superior.

Finally, in Fig. 6, we plot the curves for the channel with 6 taps. In this experiment, we can see the previous result magnified, because the longer the impulsive response of the channel is the more uncertain the channel estimate is and the higher the room for improvement for the Bayesian equalizer. In this experiment, we can observe gains over 1 dB for training sequence with 25 symbols and $\text{SNR} = 7$ dB.

V. CONCLUSIONS AND FUTURE WORK

Channel equalization has been traditionally solved using a discriminative model where the only variable modeled as random is the noise. The generative model introduced in this paper, where the posterior probability density function of the estimated CSI is included, is a more efficient solution. If we are to just estimate the encoded transmitted symbols, the discriminative model is a good choice. However, if the estimation of the APP is needed, i.e., the decoder very much benefits from this information, the discriminative solution exhibits poor results whenever the CSI is badly estimated. On the contrary, the Bayesian approach exploits the full statistical model to provide optimal APP. We prove that these estimations are useful if a LDPC encoding is used. Other decoders may take advantage of this solution as well.

Despite both the BCJR algorithm and LDPC decoding can be computed efficiently using machine learning algorithms applied on graphs [16], the drawback of this proposal is its computational complexity, because we have to compute several times the results for the BCJR algorithm to average the integral (4). Alternative graphical representation or inference algorithms that capture the essence of this approach may yield sub-optimal but less demanding solution for Bayesian equalization proposed. Furthermore, other systems and channels can be considered. width=2.5in]subfigcase1

ACKNOWLEDGMENT

This work was partially funded by Spanish government (Ministerio de Educación y Ciencia TEC2006-13514-C02-{01,02}/TCM and TEC2009-14504-C02-{01,02}, Consolider-Ingenio 2010 CSD2008-00010), and the European Union (FEDER).

REFERENCES

- [1] J. G. Proakis, *Digital Communications*, 5th ed. New York, NY: McGraw-Hill, 2008.
- [2] D. Forney, "The Viterbi algorithm," *Proc. IEEE*, vol. 61, no. 2, pp. 268–278, Mar. 1973.
- [3] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inf. Theory*, vol. 20, no. 2, pp. 284–287, Mar. 1974.
- [4] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, March 2008.
- [5] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New Jersey, USA: John Wiley & Sons, 2006.
- [6] P. M. Olmos, J. Murillo-Fuentes, and F. Perez-Cruz, "Joint nonlinear channel equalization and soft LDPC decoding with Gaussian processes," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1183–1192, Mar 2010.
- [7] M. Luby, M. Mitzenmacher, A. Shokrollahi, D. Spielman, and V. Stemann, "Efficient erasure correcting codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 569–584, Feb. 2001.
- [8] S. Chung, D. Forney, T. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Commun. Lett.*, vol. 5, no. 2, pp. 58–60, Feb. 2001.
- [9] C. Douillard, M. Jezequel, C. Berrou, A. Picart, P. Didier, and A. Gleivieux, "Iterative correction of intersymbol interference: Turbo-equalization," *European Trans. Telecomm.*, vol. 6, no. 5, pp. 507–511, Sep.-Oct. 1995.
- [10] L. M. Davis, I. B. Collings, and P. Hoeher, "Joint MAP equalization and channel estimation for frequency-selective and frequency-flat fast-fading channels," *IEEE Trans. Commun.*, vol. 49, no. 12, pp. 2106–2114, Dec. 2001.
- [11] R. Otnes and M. Tuchler, "On iterative equalization, estimation, and decoding," in *Proc. IEEE Int. Conf. on Commun.*, vol. 4, May 2003, pp. 2958–2962.
- [12] X. Wang and R. Chen, "Blind Turbo equalization in Gaussian and impulsive noise," *IEEE Trans. Veh. Technology*, vol. 50, no. 4, pp. 1092–1105, Jul. 2001.
- [13] B. Lu and X. Wang, "Bayesian blind Turbo receiver for coded OFDM systems with frequency offset and frequency-selective fading," in *Proc. IEEE Int. Conf. on Commun.*, vol. 1, Apr.-May 2002, pp. 44–48.
- [14] P. Piantanida, S. Sadough, and P. Duhamel, "On the outage capacity of a practical decoder accounting for channel estimation inaccuracies," *IEEE Trans. Commun.*, vol. 57, no. 5, pp. 1341–1350, May 2009.
- [15] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation theory*. New York, NY: Prentice-Hall, 1993, vol. 1.
- [16] F. R. Kschischang, B. I. Frey, and H. A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 498–519, Feb. 2001.